을 것이다. 그러나, 최근 모 포털 업체의 고객 이메일 데이터 유실 사건과 모 금융사의 전산망 마비 사태에서 보듯이, 완벽한 정보 시스템은 고가용성 S/W 및 H/W 인프라만으로 구축되는 것이 아니라, 높은 IT 보안 의식도 필요로 하고, 불의의 사고에 대비하여 철저한 데이터 백업도 함께 이루어 져야 한다는 것을 알 수 있다. 철저하게 안정적으로 정보 시스템을 유지한다고 해도, 사용자 및 운영자의 실수, 프로그램의 오류, 시스템 구성 변경 작업 오류 등 데이터 손실이 발생할 확률은 늘 존재하며, 이를 대비해야 하는 것이 상식인 것이다.

최적화된 백업/복구 구성안은 데이터 백업의 목적은 복구에 있음으로, 업무 데이터의 가치에 맞게끔 복구 목표 시점 및 복구 목표 시간을 설정하고, 거기에 맞게끔 백업 정책 및 백업 시스템을 구축해야만 한다. 백업 장비 선정에 있어서도, 디스크 백업과 테이프 백업 중 무엇이 더 우수하다는 우열을 가리는 어리석은 논쟁보다는, 복구 목표를 최우선 고려하고 어떤 경우에도 백업된 데이터가 없어서 복구를 필요로 하는 업무가 복구되지 못하는 경우가 없도록 꾸미는 것이, 최적화된 백업/복구 시스템을 구축하는 것이다.

일반적인 작은 규모의 데이터 손실의 경우는 메인 센터와 보조 센터의 이중 삼중의 디스크 기반 솔루션에서 복구가 될 수 있지만, 앞서 언급된 최근 사례에서도 보듯이 불의의 사고에 의해 모든 디스크 데이터가 손실되어 디스크에서 복구가 될 수 없는 경우, 테이프 백업이 없었다면 상상하기도 싫은 상황이 발생하였을 것이다. 즉, 디스크와 테이프라는 서로 다른 물리적 매체에 백업을 동시에 해두었기 때문에, 디스크로 인한 복구가 어려워지자 테이프 데이터로 복구에 성공하여 시스템을 정상화 할 수 있었던 것이다. 일반적인 경우 흔히 신규 업무 또는 변경된 업무에 대한 백업이 이루어지지 않아 복구를 할 수 없는 경우가 종종 일어난다. 데이터 보호를 위한 백업 업무는 보험과 같은 것이고, 최후의 보루인 것이다. 그럼으로, "이런 일이 일어날까?" 하는 생각보다는 "이런 일이 일어날 수도 있다"는 것에 대비하여, 철저한 복구 준비를 하는 것이 최적화된 백업/복구 방안이라고 생각한다.

COVER STORY 04

오라클 하드웨어 고급 보안 기능 – 암호화

저자 – **김일호** 수석 컨설턴트, 한국오라클 시스템 사업부 (ilho.Kim@Oracle.com)

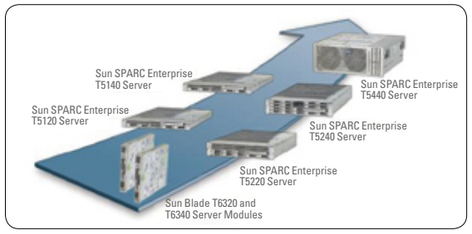
보안을 생각할 때 당연히 생각할 수 있는 부분이 바로 "암호화"다. 네트워크로 주고받는 모든 데이터들이 쉽게 Snooping되어 데이터의 내용을 열어볼 수 있거나, 해킹을 통해 유출된 개인 정보가 전혀 암호화가 되어 있지 않아, 누구나 내용을 열어볼 수 있는 일들은 당연히 데이터의 암호화를 통해 방지해야 할 것이다. 하드웨어가 보안 기능을 가진다? 바로 Oracle Server System이 가진 보안 기능은 암호화 가속 기능(On-chip cryptographic accelerator)이다.

암호화는 무거운 작업

쉽게 예상할 수 있겠지만, 암호화는 매우 CPU-intensive작업이다. 다시 말해 compute-intensive일로 암호화 자체만으로 매우 많은 CPU Time(cycle)을 소모하게 된다. 암호화란 기존의 데이터와 연계가 없도록 전혀 다른 데이터 값을 만들어내므로, 주어진 데이터를 메모리에서 읽어 암호화 알고리즘(Cipher)을 통해 값을 바꾸고, 다시 메모리에 저장하는 시스템 입장에서 매우 많은 노동이 드는 작업이다. 암호화 작업이 없는 애플리케이션이 동작하는 시스템에서 암호화 작업을 적용한다면, 기존의 시스템 자원의 많은 부분을 암호화에 사용되게 되어 당연히 전체적인 시스템 성능이 떨이지게 된다. Throughput이 낮아지며, Client/user 입장에서는 Latency가 늘어나게 되는 결과를 가져올 수밖에 없다.
몇몇 시스템 관리자, 엔지니어, 개발자들은 이러한 암호화 성능을 보완하기 위해 별도의 Cryptographic accelerator card를 사용하거나, 시스템에 따라 연산을 덜기 위한 co-processor가 존재하는 Appliance를 사용하는 경우도 있다. 그러나 당연히 매우 높은 비용이 소모되게 되며, 더구나 해당 카드를 장착하고 소프트웨어와 드라이버를 설치해야 하며, 해당 장치를 활용하기 위해 시험, 적용하는 작업이 뒤따르게 되어 고객의 입장에서 매우 큰 비용과 수고가 필요하다.

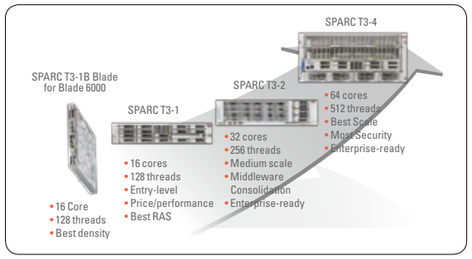
암호화 가속기가 CPU안에!

Oracle 또는 기존 Sun Microsystem의 T시리즈 시스템에 장착된 CMT Processor는 이러한 암호화 가속기가 이미 CPU core안에 On-chip되어 있다. 추가적인 비용과 수고 없이 그대로 암호화 가속기를 사용할 수 있으며, Multi-core라는 강점의 CMT Processor는 core마다 별도의 암호화 가속기를 가지고 있어, 그 성능은 Multi-processing 또는 가상화에서 더 크게 활용될 수 있다. CMT Processor는 UltraSPARC T1 à T2 à T2+ àT3로 발전해 오면서 이러한 암호화 가속기를 지속적으로 적용하고 기능을 업그레이드하고 있다.
이미 SPARC T1000, T2000, T5X20, T5X40, T3-1,2,4의 Oracle T시리즈 서버를 구입한 고객은 암호화 가속기를 가지고 있는 것으로, Solaris를 통해 쉽게 암호화 기능이 필요한 Software에서 사용할 수 있다.



<그림 1> T2, T2+ CPU가 장착된 T5X20, T5X40, T6320, T6340 서버 시스템

<그림 1>에 소개된 바로 이전 세대인 UltraSPARC T2, T2+ Processor를 사용한 시스템으로 모두 암호화 가속 기능을 지원한다.



<그림 2> 최신 T3 CPU가 장착된 T3-1,2,4 서버 시스템

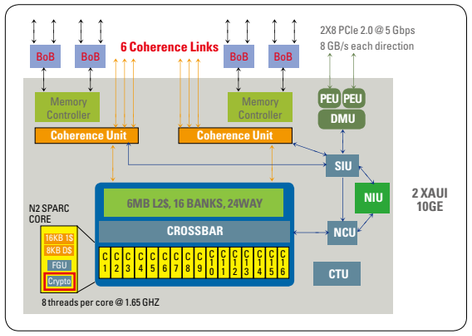
<그림 2>와 같이 최근에 발표된 T3 Processor를 사용한 시스템은 무려 CPU에 16core를 가지고 있으며, 해당 core모두 각각 암호화 가속기를 포함하고 있다. 또한 T2+에서 지원하지 않던 추가적인 알고리즘을 지원하며, 성능 또한 높아졌다.

PROCESSOR FEATUREd	ULTRASPARC T1	ULTRASPARC T2 /T2 PLUS	ULTRASPARC T3
NO. OF CORES	8	8	16
NO. OF THREADS/CORE	4	8	8
NO. OF CRYPTOGRAPHIC ACCELERATOR UNITS / PROCESSOR	8	8	16

<그림 3> UltraSPARC T Processor 별 core, thread, cryptographic accelerator 수 비교

조금 복잡한 그림일 수 있으나, <그림 4>의 실제 T3 CPU block diagram을 보면 16개의 core안에 모두 MAU(Modular Arithmetic Unit)라고 불리는 암호화 가속 Unit이 포함되어 있다

MAU는 CPU core가 암호화 연산 시 연산의 대부분을 덜어주게 되어, CPU core는 application의 다른 일을 수행할 수 있게 됨으로, 시스템은 매우 빠르게 암호화 연산이 포함된 Heavy load를 실행할 수 있다.



<그림 4> T3 CPU Block Diagram

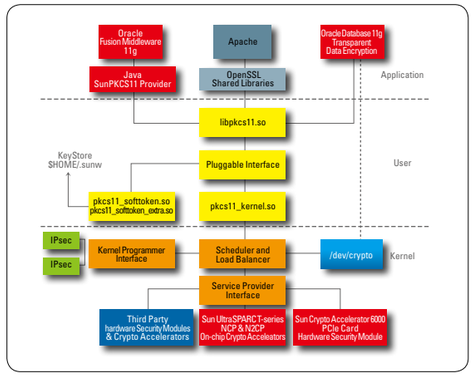
<그림 5>는 각각의 T Processor의 암호화 가속 기능이 지원하는 알고리즘과 기타 정보에 대한 것으로, 중요한 것 중 하나는 이러한 암호화 가속 기능을 Solaris에서 NCP, N2CP Driver와 SCF(Solaris Cryptographic Framework)를 통해 매우 쉽게 지원하고 있는 부분이다.

SUPPORT			
ACCELERATOR DRIVER	NPC	NCP, N2CP, N2RNG	NCP, N2CP, N2RNG
PUBLIC KEY ENCRYPTION	RSA, DSA	RSA, DSA, ECC	RSA, DSA, ECC
BULK ENCRYPTION	-	AES, DES, 3DES, RC4	AES, DES, 3DES, RC4,Kasumi
MESSAGE DIGESTS	-	MD5, SHA-1, SHA-256	MD5, SHA-1, SHA-256, SHA-512 and HMAC
APIS	PKCS#11	PKCS#11	PKCS#11
BADOM NUMBER GENERATION	-	N2RNG	N2RNG

<그림 5> T CPU 간 지원 Cipher, Hash Algorithm

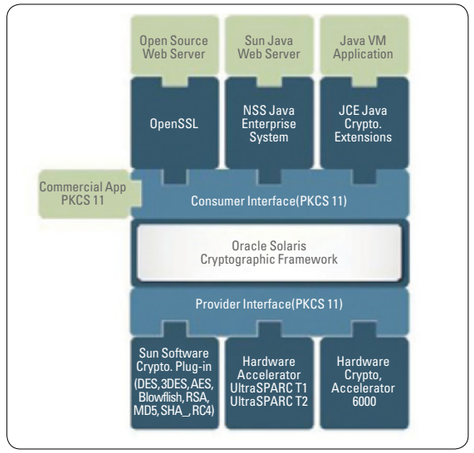
편리한 Solaris의 역할, SCF(Solaris Cryptographic Framework)

Solaris는 SCF(Solaris Cryptographic Framework)를 이용하여 User Level, Kernel Level에서 모두 암호화 가속기를 쉽게 사용할 수 있도록 기능을 제공하고 있다. 시스템에 Solaris를 설치하면 자동으로 암호화 가속기를 인식하여, 바로 사용할 수 있으며, 명령어(cryptoadm)를 제공하여, 암호화 가속 모듈을 제어하고 기능을 제어하고 관리할 수 있다.
SCF는 PKCS#11 표준을 지원하며, PKCS#11을 사용하는 Application은 쉽게 Cryptographic accelerator Hardware Provider를 사용할 수 있다.
별도의 암호화 가속기를 설치하더라도 SCF를 통해 장치를 등록된 장치를 사용하게 된다.
<그림 6>에서와 같이 SCF는 간결하고 명확한 구조의



<그림 6> SCF(Solaris Cryptographic Framework)와 각 구성요소

Framework으로 Application과 Kernel에서의 암호화 모듈을 사용을 쉽게 지원하고 있다. Oracle WebLogic Suite, Oracle Database TDE(Transparent Data Encryption), Apache SSL 등 암호화 기능을 사용하는 Application은 SCF를 통해 바로 암호화 가속기를 사용하여, Performance를 높일 수 있다.
Solaris에서 JVM 1.5 이상의 경우, JCE가 직접SCF를 사용하게 되어 있어 JCE(Java Cryptographic Extension)를 사용하는 경우 별다른 설정 없이 암호화 가속 기능을 사용할 수 있으며, Solaris에 들어있는 OpenSSL을 사용하는 PKCS#11를 이용하여 암호화 가속 기능을 역시 쉽게 사용할 수 있다. <그림 7>은 SCF의 Pluggable 구조를 보다 쉽게 설명하고 있다.



<그림 7> SCF(Solaris Cryptographic Framework) 구조

유용한 cryptoadm 명령어

Solaris는 cryptoadm이란 명령어 암호화 알고리즘 종류(mechanism으로 불림), 암호화 제공자(provider로 암호화 연산을 하는 주체)등을 모두 컨트롤 할 수 있다. SCF의 상태 정보도 확인이 가능하다.

cryptoadm 주요 기능

l Installing and uninstalling cryptographic providers
l Configuring the mechanism policy for each provider
l Displaying information about the framework

예로SPARC T3-1 서버에서 아래와 같이 실행하면 간략한 provider정보를 볼 수 있다.

```
bash-3.00# cryptoadm list

User-level providers:
Provider: /usr/lib/security/$ISA/pkcs11_kernel.so
Provider: /usr/lib/security/$ISA/pkcs11_softtoken_extra.so

Kernel software providers:
    des
    aes256
    arcfour2048
    blowfish448
    sha1
    sha2
    md5
    rsa
    swrand

Kernel hardware providers:
    ncp/0
    n2cp/0
    n2rng/0
bash-3.00#
```

Kernel hardware providers에 나타나는 ncp, n2cp, n2rng가 하드웨어 암호화 가속기를 의미한다.
아래와 같이 실행할 경우, 현재 암호화 가속기에서 지원하도록 되어 있는 암호화 알고리즘과 방식을 보여준다. RSA, DES, AES 등 많이 사용되는 Cipher 종류를 확인할 수 있다.

```
bash-3.00# cryptoadm list -m provider=ncp/0
ncp/0: CKM_DSA,CKM_RSA_X_509,CKM_RSA_PKCS,CKM_RSA_PKCS_KEY_PAIR_GEN,CKM_DH_PKCS_KEY_PAIR_GEN,CKM_DH_PKCS_DERIVE,CKM_ECDSA_KEY_PAIR_GEN,CKM_ECDH1_DERIVE,CKM_ECDSA
bash-3.00# cryptoadm list -m provider=n2cp/0
n2cp/0: CKM_DES_CBC,CKM_DES_CBC_PAD,CKM_DES_ECB,CKM_DES3_CBC,CKM_DES3_CBC_PAD,CKM_DES3_ECB,CKM_AES_CBC,CKM_AES_CBC_PAD,CKM_AES_ECB,CKM_AES_CTR,CKM_MD5,CKM_SHA_1,CKM_SHA256,CKM_MD5_HMAC,CKM_SHA_1_HMAC,CKM_SHA256_HMAC,CKM_MD5_HMAC_GENERAL,CKM_SHA_1_HMAC_GENERAL,CKM_SHA256_HMAC_GENERAL,CKM_SSL3_MD5_MAC,CKM_SSL3_SHA1_MAC,CKM_SHA384,CKM_SHA512,CKM_AES_CFB128,CKM_AES_GCM
```

암호화 알고리즘을 enable/disable 할 수 있는 기능도 가능하다.

```
# cryptoadm disable provider=/usr/lib/security/\$ISA/pkcs11_softtoken.so mechanism=CKM_AES_CBC_PAD
# cryptoadm enable provider=n2cp/0 mechanism=CKM_AES_CBC_PAD
```

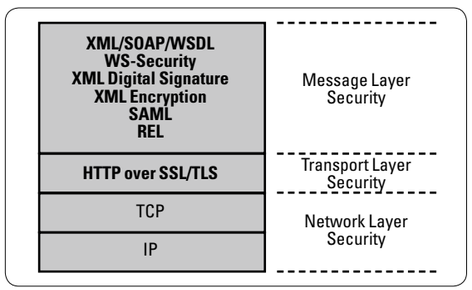
암호화 가속기 활용

-Secured Web service

SSL/TLS을 활용하는 고객은 누구보다 SPART T시리즈 시스템을 쉽게 활용할 수 있다. 쉬운 예로 Apache web server 또는 Sun Java System Web Server를 고객이 보안을 위해, http://hostname이 아닌 https://hostname로 접속으로 변경한다면, 매번 발생하는 모든 트랜잭션의 SSL/TLS 작업은 많은 시스템 리소스를 사용하겠지만, 암호화 가속기를 활용함으로써 그러한 CPU 로드를 덜어 매우 큰 성능 향상을 가져올 수 있다. 가속기가 없는 다른 시스템이라면, 성능을 위해 시스템을 증설하거나, 비싼 암호화 가속 카드를 구입을 지불해야만 한다.
국내의 많은 사이트들은 현재 https를 사용하지 않고 있어, 암호화 가속 부분 지원은 큰 이점이 될 수 있다.

-Oracle WebLogic Applications

Middleware software인 WebLogic의 경우, Transaction과 데이터의 암호화 모두를 사용할 경우 암호화 가속기를 이용하여 쉽게 성능을 향상 시킬 수 있다. 실제 Transport Layer에서 SSL/TSL 암호화를 통해 데이터를 보호할 수 있지만, Proxy를 지난 데이터 레벨까지 보안을 위해서는 Message Layer까지 암호화를 적용해야만 한다.
높은 수준의 보안을 위해 암호화를 적용해야 하는 고객의 경우, 역시 이러한 과정에서 큰 Advantage를 가질 수 있다. 〈그림 8〉의 각 Layer별도로 또는 모두 암호화 기능을 활용할 수 있다. 설정하는 부분의 아래의 가이드북을 참조하여 설정을 할 수 있다.
(OracleFusionMiddleware-SecuringWebLogicWebServicesforOracleWebLogicServerguide)



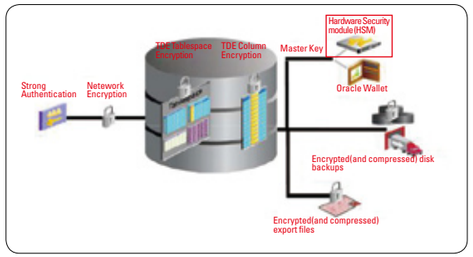
<그림8> Weblogic의 암호화 적용 Layers

-Java Application

기존 Sun JCE(Java Cryptographic Extension)는 JCA(Java Cryptography Architecture)로 통합되었다. JCA를 활용하는 애플리케이션의 경우, 매우 편리하게 암호화 가속기를 사용할 수 있다. 간단한 Property 변경만으로 JCA가 하드웨어 암호화 가속기를 이용할 수 있기 때문이다.
(http://download.oracle.com/javase/6/docs/technotes/guides/security/crypto/CryptoSpec.html)

Oracle TDE(Transparent Data Encryption)

최근 보안 이슈로 DB 데이터의 암호화 중요성이 더 커지고 있다. Oracle Database Enterprise Edition의 TDE는 Table과 Column을 데이터를 Encryption/Decryption 하는 보안 기능을 제공한다. TDE의 암호화 작업이 암호화 가속기를 사용함으로써 성능 향상을 가져올 수 있다.
아래는 TDE에서 지원하는 암호화 알고리즘으로 모두 암호화 가속기가 지원하고 있다.



<그림 9> TDE 구조와 암호화 가속 모듈

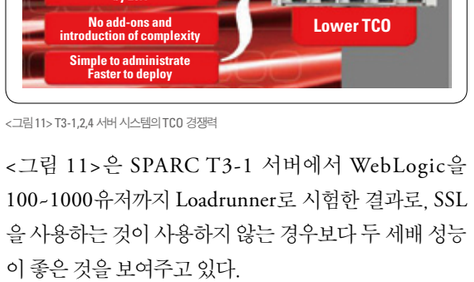
Oracle TDE(Transparent Data Encryption)

최근 보안 이슈로 DB 데이터의 암호화 중요성이 더 커지고 있다. Oracle Database Enterprise Edition의 TDE는 Table과 Column을 데이터를 Encryption/Decryption 하는 보안 기능을 제공한다. TDE의 암호화 작업이 암호화 가속기를 사용함으로써 성능 향상을 가져올 수 있다. 아래는 TDE에서 지원하는 암호화 알고리즘으로 모두 암호화 가속기가 지원하고 있다.

Algorithm	Key Size	Parameter Name
Triple DES (Data Encryption Standard)	168 bits	3DES168
AES (Advanced Encryption Standard)	128 bits	AES128
AES	192 bits (default)	AES192
AES	256 bits	AES256

뛰어난 성능

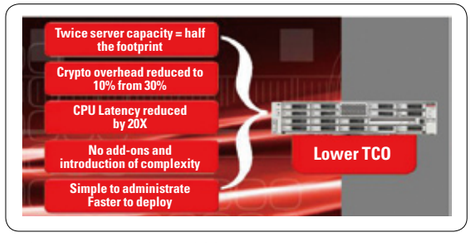
System	Configuration	Result
Sun SPARC Enterprise T5220 Server	One processor with eight cores running at 1.4GHz	37,001
Sun SPARC Enterprise T2000 Server	One processor with eight cores running at 1.4GHz	16,407
Sun Fire T2000 Server	One processor with eight cores running at 1.2GHz	14,001
Sun Fire T1000 Server	One processor with eight cores running at 1.2GHz	10,466
Powerleader Science & Technology PR4700D Server	Two processors with eight cores running at 3.0GHz(Intell)	22,332
HP ProLiant DL580 G5 Server	Four processors with eight cores running at 2.4GHz(Intell)	26,119
HP ProLiant DL580 G4 Server	Four processors with 16 cores running at 2.93GHz(Intell)	30,261
HP ProLiant DL580 G2 Server	Four processors with eight cores running at 3.9GHz(AMD)	22,254
Dell Power Edge 2950 Server	wo processors with eight cores running at 3.66GHz(Intell)	16,830
Dell Power Edge 2950 Server	wo processors with four cores running at 3.0GHz(Intell)	14,495
Fujitsu Primergy Tx300 S3 Server	wo processors with eight cores running at 3.66GHz(AMD)	18,160
Fujitsu Primergy Tx600 S3 Server	wo processors with eight cores running at 3.4GHz(Intell)	14,896



<그림 10> SPECweb2005 공인 성능수치

암호화 가속기는 CPU에 built-in된 구조로 매우 높은 성능을 낸다. 아래의 성능 그래프가 그러한 사실을 증명하고 있다. 높은 Clock speed를 가진 X86 시스템과 비교해도 그 성능이 훨씬 높으며, 암호화 연산이 얼마나 많은 CPU 자원을 소모하는지 반증하는 결과로도 볼 수 있다.

<그림 10>은 spec.org에서 시험한 SPECweb2005 공인 성능치의 자료로 SPECWeb2005는 SSL을 이용한 트랜잭션이 벤치마크에서 활용된다. 동일 코어 수에서 CPU Clock의 큰 차이에도 전체적인 벤치마크 수치는 타 시스템보다 매우 높은 것을 확인할 수 있다.



<그림 11> T3-1,2,4 서버 시스템의 TCO 경쟁력

<그림 11>은 SPARC T3-1 서버에서 WebLogic을 100~1000유저까지 Loadrunner로 시험한 결과로, SSL을 사용하는 것이 사용하지 않는 경우보다 두 세배 성능이 좋은 것을 보여주고 있다.

결론

고객은 SPARC T시리즈 시스템을 도입하게 되면, 강력한 암호화 가속 기능은 그대로 부가적인 비용 없이 활용이 가능하다. 앞으로 시스템의 여러 장점 중 강력한 무기가 될 것이다. 더구나 Oracle의 암호화 기능을 가진 Software들은 Oracle on Oracle의 모토대로 가속 기능을 바로 활용이 가능하다. 다른 Vendor가 제공할 수 없는 큰 경쟁력이다.

보안 이슈가 점점 많아지는 상황에서도 인터넷을 사용하면서 국내 사이트 중 https로 접속하는 곳은 그렇게 많지 않다. 또한 고객 정보가 유출되었다는 뉴스에서 데이터가 암호화 되어 문제가 되지 않는다는 이야기는 듣기 어렵다. 하지만, 앞으로 IT에서 암호화 기술의 적용은 기업이나 개인의 입장에서도 꼭 필요하며, 그 중요성이 부각되면서 계속적으로 빠르게 적용해 나갈 것이다. 그러한 시점에서 이미 2005년부터 개발된 Oracle T시리즈 시스템의 준비된 암호화 가속 기술은 Oracle과 고객 모두에게 큰 선물이 될 것이다.

참조

-URLs
http://www.coresecuritypatterns.com/blogs/?tag=ws-security
http://www.oracle.com/technetwork/articles/systems-hardware-architecture/security-webLogic-t-series-168447.pdf
http://www.oracle.com/technetwork/articles/systems-hardware-architecture/hi-perf-soa-xml-svcs-172821.pdf
http://blogs.oracle.com/sprack/
http://www.oracle.com/technetwork/database/options/advanced-security/ds-security-advanced-security-11gr2-1-129479.pdf
http://www.pki.gov.sa/workshop/speakers/presentation/Oracle-1.pdf

-Documents
BluePrints: USING THE CRYPTOGRAPHIC ACCELERATORS IN THE ULTRASPARC® T1 AND T2 PROCESSORS
BluePrints: taking advantage of wire-speed Cryptography In Oracle WebLogic Server 10.3.x and Java™ Platform, Enterprise Edition 5 Application Environments
Oracle White paper: Oracle Solaris and Sun SPARC Systems—Integrated and Optimized for Enterprise Computing

-Slides
Crypto on the T3 - Chad Prucha, Oracle Corporation
Product Essentials for Sales Consultants SPARC T3 Systems – Oracle Corporation