Security in knowledge

What? Me, Worry? I've Already Been Hacked. Haven't You?

David Maman
Co-Founder, CTO
GreenSQL

RSA CONFERENCE 2013

Session ID: DSP-F43
Session Classification: General Interest

#1 Global Security Challenge

Sophisticated attacks: - executed by financially-motivated cybercriminals, state- sponsored espionage groups, hacktivists and insiders/Privileged users

Information theft – In 2011 174M Records Exposed in the USA. The average cost of a data breach **$214** per record (Source: DataLossDB, Ponemon Institute, 2010, Verizon - 2012 Data Breach Investigations Report).

Compliance - 96% of victims subject to PCI Dss had not achieved compliance (Source: Verizon - 2012 Data Breach Investigations Report)

FBI - Organized data theft bigger criminal industry than drug trading (source FBI)

2011 database servers accounted for 96% of all records breached.
(Source: Verizon - 2012 Data Breach Investigations Report)

greensql
Database Security Made Simple

It’s not getting any better…

- Jun 2012 - **Amazon’s Zappos.com -** data breach exposed personal information of 24 million customers. (Amazon acquired Zappos for more than $1 billion.)
- Feb 2012 - **University of North Carolina**, 350,000 records, exposed Social Security numbers and financial information online
- March 2012 - **VISA and MasterCard** alerted banks about a recent breach at a **Global Payment Systems.** 7 million records, including 1.5 million credit cards
- April 2012 **- South Carolina Health and Human Services** 228,435 records of patient records, including Medicaid ID numbers

greensql
Database Security Made Simple

It’s not getting any better…

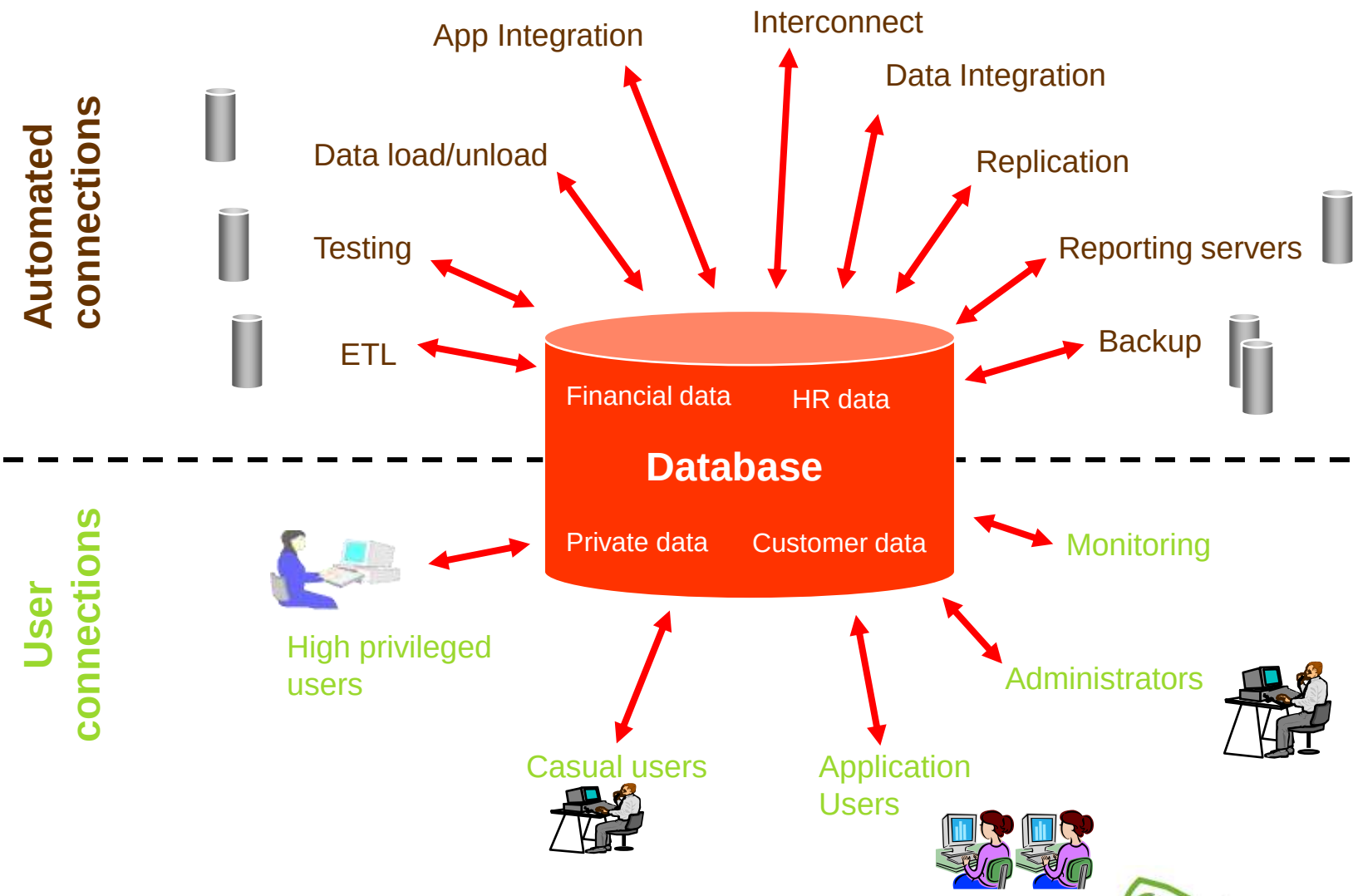
- May 2012 **University of Nebraska breach -** 654,000 Social Security numbers, addresses, grades, financial aid information for current and former NU students
- May 2012 – **University of Nebraska** Breach Highlights Education In Crosshairs Database containing 654,000 exposed
- June 2012 - **LinkedIn** reported a breach of at least **6.5 Million** passwords.
- July 2012 – **KT Mobile**(South Korea) **8.7 million** customer records stolen
- Sep 2012 - **Saudi Arabia's national oil company** Most of the databases
- Nov 2012 - **Adobe** 150,000 User Accounts Exposed

greensql
Database Security Made Simple

Motivation for Database Security

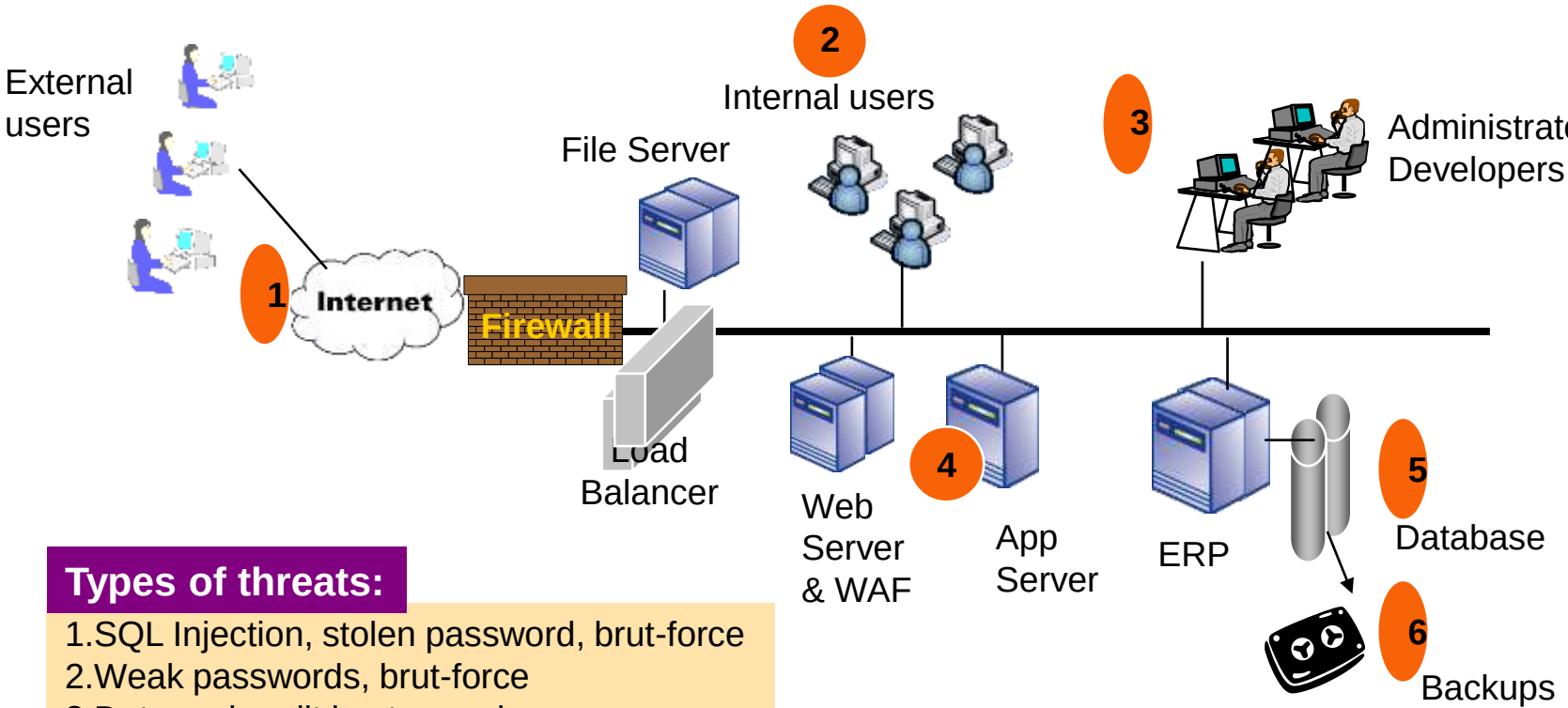
Security in knowledge

Who uses the Database?



greensql
Database Security Made Simple

Threats..



Types of threats:

1. SQL Injection, stolen password, brut-force
2. Weak passwords, brut-force
3. Data and audit log tampering
4. Vulnerabilities, password exposed
5. Weak AAA and database security
6. Tapes stolen/lost

Insider threats a concern:

80% of threats come from insiders
75% of internal threats are undetected
25% of enterprises detected security breaches
60% of data loss/corruption caused by human error

greensql
Database Security Made Simple

Information Theft

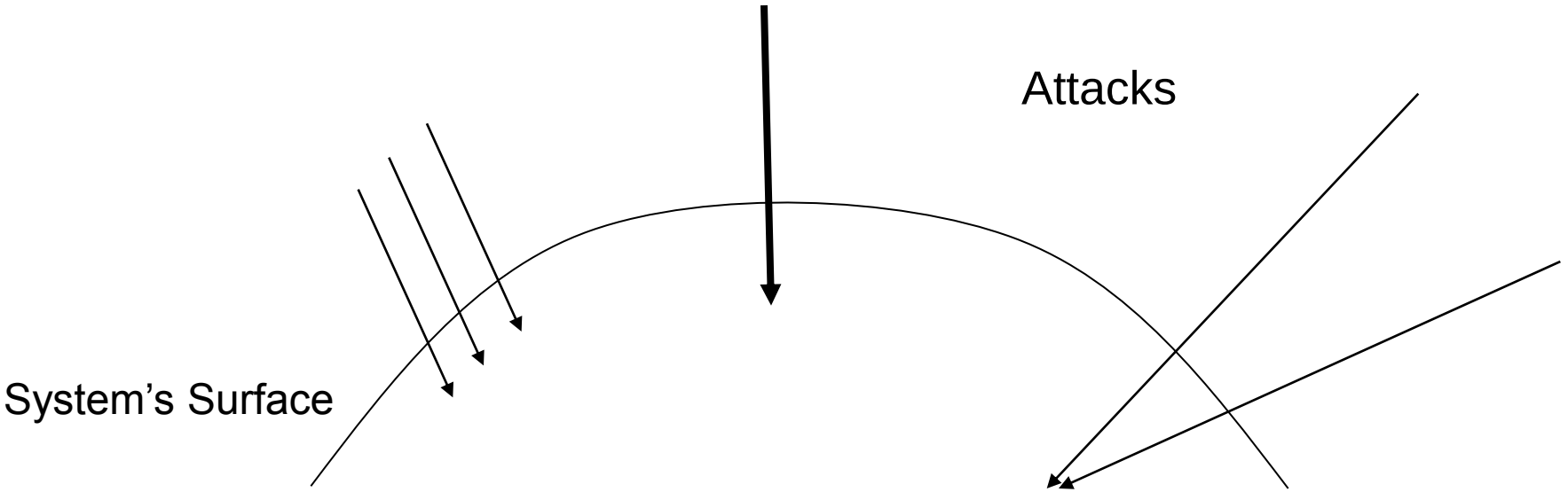
- Hackers have become professional
- Business models finance them
- SQL Injection attacks are becoming increasingly sophisticated and difficult to prevent
- SQL Injections discovered in “close source” apps
- It uses stealth techniques to go unnoticed for as long as possible
- Database attacks are getting to be more and more valuable for hackers and Crime organizations

Pricelist

Address	$0.50
Phone number	$0.25
Unpublished phone number	$17.50
Cell phone number	$10
Date of birth	$2
Social Security number	$8
Driver's license	$3
Education	$12
Credit history	$9
Bankruptcy details	$26.50
Lawsuit information	$2.95
Sex offender	$13
Workers' comp history	$16
Military record	$35

greensql
Database Security Made Simple

Attackability



How to reduce the ways attackers can penetrate surface

greensql
Database Security Made Simple

Windows Attack Vectors

- Open sockets
- Open RPC endpoints
- Open named pipes
- Services
- Services running by default
- Services running as SYSTEM
- Active Web handlers
- Active ISAPI Filters
- Dynamic Web pages
- Executable vdirs
- Enabled accounts
- Enabled accounts in admin group
- Null Sessions to pipes and shares
- Guest account enabled
- Weak ACLs in FS
- Weak ACLs in Registry
- Weak ACLs on shares
- *VBScript enabled*
- *Jscript enabled*
- *ActiveX enabled*
- *Third party application*

greensql
Database Security Made Simple

What is SQL Injection?

- SQL Injections attacks goes directly after your most valuable asset – The Database
- Uses the same connectivity as legitimate web and other application usage
- Network and OS security won't help
- Many systems and close source applications vulnerable, even among the big names
- Extremely easy to learn / attempt
- Endless amount of information is available

greensql
Database Security Made Simple

How common is SQL Injection?

- It is probably the most common vulnerability today!
- SQL Injections discovered in Close source applications!
- It is a flaw in "web application" development, it is not a database or web server problem
  - Most programmers are still not aware of this problem
  - A lot of the tutorials & demo “templates” are vulnerable
  - Even worse, a lot of solutions posted on the Internet are not good enough
- Some pen tests at the wild shows over 65% of clients turn out to be vulnerable to some sort SQL Injection attacks.

greensql
Database Security Made Simple

What is SQL?

"Users" Table

UserName	FirstName	LastName	Password
CJONES	Cynthia	Jones	XXXXXX
BSMITH	Bill	Smith	YYYYYY
SKING	Susan	King	ZZZZZZZ
RSMITH	Rob	Smith	AAAAA

Column data returned

```
SELECT UserName, Password
FROM Users
WHERE LastName = 'Smith'
```

Table containing data

Criteria rows must meet

UserName	Password
BSMITH	YYYYYY
RSMITH	AAAAA

Query Results

greensql
Database Security Made Simple

The Trick

- SQL statements created by concatenating SQL code fragments with user-supplied values
- What if user-supplied values were constructed to contain SQL code fragments that changed the meaning of the statement?
- What if we could turn it into a statement that matched records without matching on the username and password, as was intended?

greensql
Database Security Made Simple

Vulnerable Applications

- Almost all SQL databases and programming languages are potentially vulnerable
  - MS SQL Server, Oracle, MySQL, Postgres, DB2, MS Access, Sybase, Informix, etc

- Accessed through applications developed using:
  - Perl and CGI scripts that access databases
  - ASP, JSP, PHP
  - XML, XSL and XSQL
  - Javascript
  - VB, MFC, and other ODBC-based tools and APIs
  - DB specific Web-based applications and API's
  - Reports and DB Applications
  - 3 and 4GL-based languages (C, OCI, Pro*C, and COBOL)
  - many more

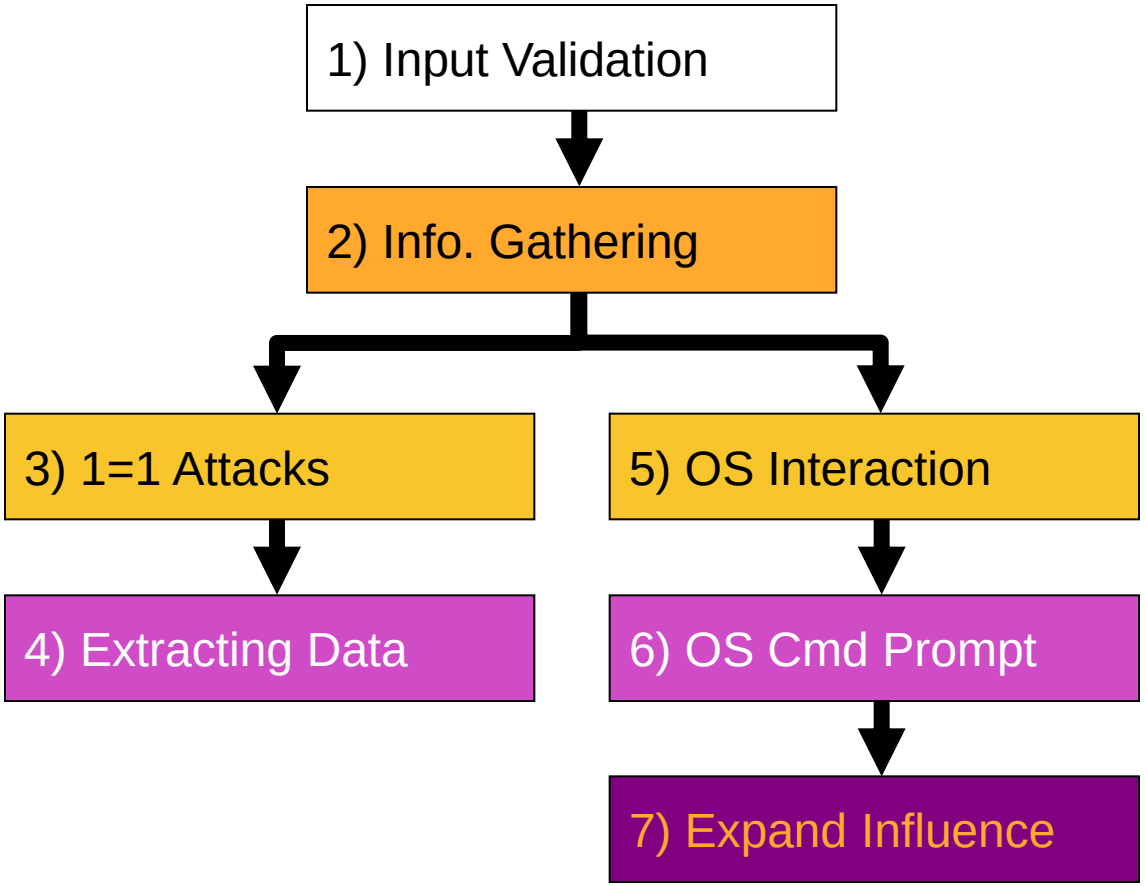
greensql
Database Security Made Simple

SQL Injection Characters

- ' or " character String Indicators
- -- or # single-line comment
- /*…*/ multiple-line comment
- + addition, concatenate (or space in url)
- || (double pipe) concatenate
- % wildcard attribute indicator
- ?Param1=foo&Param2=bar URL Parameters
- PRINT useful as non transactional command
- *@variable* local variable
- *@@variable* global variable
- waitfor delay '0:0:10‘ time delay

greensql
Database Security Made Simple

SQL Injection Methodology



greensql
Database Security Made Simple

SQL Injection results

- Bypass login page
- DOS - Deny of service
- Install web shell
- Iframe injection
- Access system files
- Install db backdoor
- Theft of sensitive information / credit cards
- Additional step of the attack:
  - Attack additional servers and computers on the LAN

greensql
Database Security Made Simple

The Database Layers

Applications

SQL Database Grammar

Database Application protocol layer (TDS, MySQL)

Connections Layer

Database

greensql
Database Security Made Simple

WAF vs. Database security

- Web Application Firewall solutions are providing security to the Web Application layer
- HTTP traffic is inspected and enforced by the Web Application Firewall.
- Over the HTTP traffic you can only try and detect signatures for SQL injection attacks.
- WAF is a very important part of your security!
- Database Firewall analyses SQL!
- A true database architecture and policy is enforced inside the Database Firewall!

greensql
Database Security Made Simple

What do I need to do..

Security

- Stops SQL Injection attacks
- Separation Of Duties
- Database firewall
- Intrusion detection/prevention
- Virtual Patching

Performance

- Protection from denial of service
- Reduce Network Connections
- Increasing Database efficiency
- Increase user experience

Auditing

- DAM (Database Activity Monitoring)
- Advanced Auditing – Before/After
- Up to a column level policy
- PCI-DSS,SOX,HIPPA reports
- Email/SYSLOG Alerts

Masking

- Up to a column level masking
- Mask per user, IP or application
- Hide sensitive data

(Credit Card/ Salaries/ Medical Information/ Other)

greensql
Database Security Made Simple

SQL Injection prevention

GreenSQL Take: SQL injection/Attacks preventions

- Enforcing Database Firewall!
- Enforcing Separation of Duties!
- Enforcing Risk Based polices!
- Enforcing Masking on sensitive information!
- Enforcing Learning Mode with SQL Injection query grouping!
- Enforcing Auditing to Administrative commands and Auditing to sensitive information access!

greensql
Database Security Made Simple

Database in the Cloud

As everything else, the database is migrating to the cloud

- Installed on a dedicated VPS with the application server.
- Installed on a dedicated server inside the private cloud serving application servers
- Database as a service (Microsoft SQL Azure, etc)

All Database needs protection, even more when using a cloud service!

greensql
Database Security Made Simple

Thank You!

David Maman – dmaman@greensql.com

For more information please visit

http://www.greensql.com

Security in knowledge